

## Mysteries of Photography #8: Binary Numbers and Photography

There is this old IT joke:

**“There are **10** types of people in the world.**

**Those who understand binary numbers**

**and those who don’t.”**

As a photographer you might ask why would you waste your time reading about binary numbers? Well, the answer is obvious. Unless you are very eccentric or trying to prove a point (or both), you are using a digital camera. Your camera is using binary numbers to quantify the light captured by every pixel in your camera and using binary numbers for all output images.

Every digital camera, even the cheapest one on the market, contains a computer with more processing power than NASA had when it sent men to the moon. All in your hand, running off a tiny battery capable of capturing 10 or more images every second.

All information flow in computers consists of ones (1) and zeros (0). Either there is a voltage signal (ones) or no voltage (zeros).

So let’s compare binary numbers with our familiar commonplace number system based on the number of fingers on both hands.

| Base 10 | Binary |
|---------|--------|
| 0       | 0      |
| 1       | 1      |
| 2       | 10     |
| 3       | 11     |
| 4       | 100    |
| 5       | 101    |
| 6       | 110    |
| 7       | 111    |
| 8       | 1000   |

From this you can see every time you add a zero onto the end of a binary number the “normal” number increases by a factor of 2 (twice as big).

Now your camera will use binary numbers that are 12 or 14 ones and zeros long (a.k.a “Bits”). For instance, the Sony a7 series uses numbers that are 14 bits in size unless you are using fast burst mode of 10 or more images per second where it drops back to 12 bits. The latest Fujifilm GFX uses 16 bit numbers in its image capture.

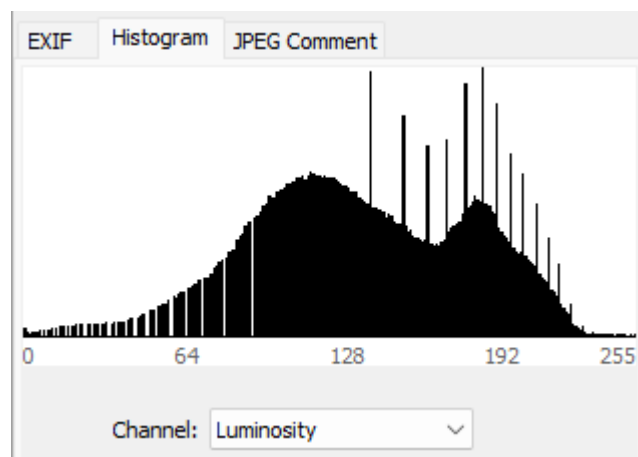
Now let’s see what this means in familiar base 10 numbers.

| Base 10 | Binary            | No. of Bits |
|---------|-------------------|-------------|
| 2       | 10                | 1           |
| 4       | 100               | 2           |
| 8       | 1000              | 3           |
| 16      | 10000             | 4           |
| 32      | 100000            | 5           |
| 64      | 1000000           | 6           |
| 128     | 10000000          | 7           |
| 256     | 100000000         | 8           |
| 512     | 1000000000        | 9           |
| 1024    | 10000000000       | 10          |
| 2048    | 100000000000      | 11          |
| 4096    | 1000000000000     | 12          |
| 8192    | 10000000000000    | 13          |
| 16384   | 100000000000000   | 14          |
| 32768   | 1000000000000000  | 15          |
| 65536   | 10000000000000000 | 16          |

Now this is like some cute Christmas tree but what relevance is this to your photography?

Firstly, remember that the JPG image format uses 8 bit numbers (256 levels). This is all very well for an end product but, if you shoot in JPG (not RAW) and if you start editing, bringing down the top white level and bringing up the bottom black, this is going to stretch the image to fill the 256 final levels and you can wind up with posterisation if you do it too much.

This is the histogram of an image probably processed in this fashion. In this case, posterisation was not obvious to the eye but was obvious in the histogram.



By contrast, if you shoot RAW, you have over 4,000 discrete levels to start with 12 bit images and over 16,000 in with 14 bit captures. This leaves you plenty of room to muck around before you compress them down to the final 256 level 8 bit JPG final product.

*So the take home message:* Photographs live and die by binary numbers and **don't shoot in JPG, use RAW!** (And now do you get the joke?)

David Woodcock GMAPS EFIAP/b